



# UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE  
United States Patent and Trademark Office  
Address: COMMISSIONER FOR PATENTS  
P.O. Box 1450  
Alexandria, Virginia 22313-1450  
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
10/673,651	09/29/2003	Shyh-Kwei Chen	YOR920030265US1	2352

28211 7590 04/11/2006

FREDERICK W. GIBB, III  
GIBB INTELLECTUAL PROPERTY LAW FIRM, LLC  
2568-A RIVA ROAD  
SUITE 304  
ANNAPOLIS, MD 21401

EXAMINER

COLAN, GIOVANNA B

ART UNIT PAPER NUMBER

2162

DATE MAILED: 04/11/2006

Please find below and/or attached an Office communication concerning this application or proceeding.

# Office Action Summary

Application No.

10/673,651

Applicant(s)

CHEN ET AL.

Examiner

Giovanna Colan

Art Unit

2162

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --

## Period for Reply

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE 3 MONTH(S) OR THIRTY (30) DAYS, WHICHEVER IS LONGER, FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133). Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

## Status

- 1) ☒ Responsive to communication(s) filed on 29 September 2003.
- 2a) ☐ This action is **FINAL**. 2b) ☒ This action is non-final.
- 3) ☐ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

## Disposition of Claims

- 4) ☒ Claim(s) 1-50 is/are pending in the application.
- 4a) Of the above claim(s) \_\_\_\_\_ is/are withdrawn from consideration.
- 5) ☐ Claim(s) \_\_\_\_\_ is/are allowed.
- 6) ☒ Claim(s) 1-50 is/are rejected.
- 7) ☐ Claim(s) \_\_\_\_\_ is/are objected to.
- 8) ☐ Claim(s) \_\_\_\_\_ are subject to restriction and/or election requirement.

## Application Papers

- 9) ☐ The specification is objected to by the Examiner.
- 10) ☒ The drawing(s) filed on 29 September 2003 is/are: a) ☒ accepted or b) ☐ objected to by the Examiner.  
Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).  
Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).
- 11) ☐ The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

## Priority under 35 U.S.C. § 119

- 12) ☐ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).
- a) ☐ All b) ☐ Some \* c) ☐ None of:
1. ☐ Certified copies of the priority documents have been received.
2. ☐ Certified copies of the priority documents have been received in Application No. \_\_\_\_\_.
3. ☐ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).
- \* See the attached detailed Office action for a list of the certified copies not received.

## Attachment(s)

- 1) ☒ Notice of References Cited (PTO-892)
- 2) ☐ Notice of Draftsperson's Patent Drawing Review (PTO-948)
- 3) ☐ Information Disclosure Statement(s) (PTO-1449 or PTO/SB/08)  
Paper No(s)/Mail Date \_\_\_\_\_
- 4) ☐ Interview Summary (PTO-413)  
Paper No(s)/Mail Date. \_\_\_\_\_
- 5) ☐ Notice of Informal Patent Application (PTO-152)
- 6) ☐ Other: \_\_\_\_\_

### DETAILED ACTION

1. This action is issued in response to applicant filed application on 09/29/2003.
2. Claims 1 – 50 are pending.

### ***Claim Rejections - 35 USC § 112***

3. The following is a quotation of the second paragraph of 35 U.S.C. 112:

The specification shall conclude with one or more claims particularly pointing out and distinctly claiming the subject matter which the applicant regards as his invention.

4. Claim 7, 14, 21, 28, 35, 42, and 49 are rejected under 35 U.S.C. 112, second paragraph, as being indefinite for failing to particularly point out and distinctly claim the subject matter which applicant regards as the invention.

The citation “ if a predicate interval is larger” in claims 7, 14, 21, 28, 35, 42, and 49, are vague includes a vague term, such as, “if” which renders the claims indefinite. Examiner is unable to assert what the method comprises when the limitation “if a predicate interval is larger” is not met.

The term “excess length” in claims 7, 14, 21, 28, 35, 42, and 49, is not clearly defined in the claim, the specification does not provide a clear standard for ascertaining what the invention entails, and one of ordinary skill in the art would not be reasonable apprised of the scope of the invention. Examiner is unable to determine what the invention entails because the term is not clearly defined in the specification.

**Examiner asserts that all claims should be checked for clarification.**

**Appropriate action is required.**

***Claim Rejections - 35 USC § 102***

5. The following is a quotation of the appropriate paragraphs of 35 U.S.C. 102 that form the basis for the rejections under this section made in this Office action:

A person shall be entitled to a patent unless –

(b) the invention was patented or described in a printed publication in this or a foreign country or in public use or on sale in this country, more than one year prior to the date of application for patent in the United States.

6. Claim 1 – 6, 8 – 13, 15 – 20, 22 – 27, 29 – 34, 36 – 41, 43 – 48, and 50 are rejected under 35 U.S.C. 102(b) as being anticipated by Reiner et al. (Reiner hereinafter) (US Patent No. 6,289,334 B1, issued: September 11, 2001).

Regarding Claim 1, Reiner discloses a method for maintaining and using a query index, wherein queries within said query index have predicate intervals (Col. 10, lines 56 – 61, Reiner), said method comprising:

defining groups of virtual construct intervals (Col. 2 and 8, lines 65 – 67 and 37 – 40; respectively, Reiner); and

inserting each of said predicate intervals into at least one of said groups of said virtual construct intervals (Col. 9, lines 12 – 13 and 27 – 28, Reiner).

Regarding Claim 2, Reiner discloses a method, wherein each of said groups of virtual construct intervals is adapted to hold multiple predicate intervals, and

wherein said groups of virtual construct intervals have uniform lengths (Col. 25, lines 44 – 47, Reiner), and

wherein said predicate intervals have non-uniform lengths (Col. 25, lines 39 – 40, Reiner).

Regarding Claim 3, Reiner discloses a method, further comprising maintaining locations of said predicate intervals within said groups of virtual construct intervals using a predicate ID bitmap vector (Col. 63, lines 36 – 38, Reiner<sup>1</sup>).

Regarding Claim 4, Reiner discloses a method, wherein said process of defining said groups of virtual construct intervals comprises beginning all virtual construct intervals in a group of virtual construct intervals at the same attribute value (Col. 10, lines 50 – 55, Reiner<sup>2</sup>) and ending said virtual construct intervals in said group of virtual construct intervals at different attribute values (Col. 9, lines 17 – 28, Reiner<sup>3</sup>).

Regarding Claim 5, Reiner discloses a method, wherein all of said groups of said virtual construct intervals within said query index have the same pattern of different sized of virtual construct intervals (Col. 4 and 25, lines 34 – 37 and 39 – 40; respectively, Reiner).

---

<sup>1</sup> Examiner interprets the pointer to the bitmap as an ID bitmap vector claimed. Reiner further discloses that this pointer is used to indicate which buffers are full. This implies the step of locating.

<sup>2</sup> Wherein same key value corresponds to same attribute value claimed. In addition, the step of simultaneously indexing in accord to the same key value corresponds to beginning the predicates at the same attribute value claimed.

<sup>3</sup> The two queries, before insertion and after insertion of predicates (disclosed by Reiner), begin with the same attribute values, such as, "SELECT name". Regarding the same citation, Reiner discloses ending the two queries at different attribute values, such as, "WHERE department" and "employee.rowid<0.0.2"

Regarding Claim 6, Reiner discloses a method, wherein said process of inserting said predicate intervals into said virtual construct intervals, comprises inserting said predicate interval into the same sized virtual construct interval (Col. 13, lines 55 – 57, Reiner<sup>4</sup>).

Regarding Claim 8, Reiner discloses a method, wherein event values of said predicate intervals are aligned with the same event values of said virtual construct intervals, such that the same event values of multiple predicate intervals are inserted into the same event value within virtual construct interval (Col. 9, lines 36 – 39, Reiner<sup>5</sup>).

Regarding Claim 9, Reiner discloses a method for maintaining and using a query index, wherein queries within said query index have predicate intervals, said method comprising:

defining groups of virtual construct intervals (Col. 2 and 8, lines 65 – 67 and 37 – 40; respectively, Reiner); and

inserting each of said predicate intervals into at least one of said groups of said virtual construct intervals (Col. 9, lines 12 – 13 and 27 – 28, Reiner),

wherein each of said groups of virtual construct intervals is adapted to hold multiple predicate intervals (Col. 2, lines 66 – 67, Reiner),

---

<sup>4</sup> Wherein examiner interprets the step of appending partition-matching predicates as a step of inserting predicate intervals into the same sized virtual construct claimed. In other words, matching the partitions implies matching the size.

<sup>5</sup> Wherein examiner interprets the step of aligning with the same event values as the step of appending a predicate for matching records claimed.

wherein said groups of virtual construct intervals have uniform lengths (Col. 25, lines 44 – 47, Reiner), and

wherein said predicate intervals have non-uniform lengths (Col. 25, lines 39 – 40, Reiner).

Regarding Claim 10, Reiner discloses a method, further comprising maintaining locations of said predicate intervals within said groups of virtual construct intervals using a predicate ID bitmap vector (Col. 63, lines 36 – 38, Reiner<sup>6</sup>).

Regarding Claim 11, Reiner discloses a method, wherein said process of defining said groups of virtual construct intervals comprises beginning all virtual construct intervals in a group of virtual construct intervals at the same attribute value (Col. 10, lines 50 – 55, Reiner<sup>7</sup>) and ending said virtual construct intervals in said group of virtual construct intervals at different attribute values (Col. 9, lines 17 – 28, Reiner<sup>8</sup>).

Regarding Claim 12, Reiner discloses a method, wherein all of said groups of said virtual construct intervals within said query index have the same pattern of different

---

<sup>6</sup> Examiner interprets the pointer to the bitmap as an ID bitmap vector claimed. Reiner further discloses that this pointer is used to indicate which buffers are full. This implies the step of locating.

<sup>7</sup> Wherein same key value corresponds to same attribute value claimed. In addition, the step of simultaneously indexing in accord to the same key value corresponds to beginning the predicates at the same attribute value claimed.

<sup>8</sup> The two queries, before insertion and after insertion of predicates (disclosed by Reiner), begin with the same attribute values, such as, "SELECT name". Regarding the same citation, Reiner discloses ending the two queries at different attribute values, such as, "WHERE department" and "employee.rowid<0.0.2"

sized of virtual construct intervals (Col. 4 and 25, lines 34 – 37 and 39 – 40; respectively, Reiner).

Regarding Claim 13, Reiner discloses a method, wherein said process of inserting said predicate intervals into said virtual construct intervals, comprises inserting said predicate interval into the same sized virtual construct interval (Col. 13, lines 55 – 57, Reiner<sup>9</sup>).

Regarding Claim 15, Reiner discloses a method, wherein event values of said predicate intervals are aligned with the same event values of said virtual construct intervals, such that the same event values of multiple predicate intervals are inserted into the same event value within virtual construct interval (Col. 9, lines 36 – 39, Reiner<sup>10</sup>).

Regarding Claim 16, Reiner discloses a method for maintaining and using a query index, wherein queries within said query index have predicate intervals, said method comprising:

defining groups of virtual construct intervals (Col. 2 and 8, lines 65 – 67 and 37 – 40; respectively, Reiner); and

---

<sup>9</sup> Wherein examiner interprets the step of appending partition-matching predicates as a step of inserting predicate intervals into the same sized virtual construct claimed. In other words, matching the partitions implies matching the size.

<sup>10</sup> Wherein examiner interprets the step of aligning with the same event values as the step of appending a predicate for matching records claimed.



inserting each of said predicate intervals into at least one of said groups of said virtual construct intervals (Col. 9, lines 12 – 13 and 27 – 28, Reiner),

wherein each of said groups of virtual construct intervals is adapted to hold multiple predicate intervals (Col. 2, lines 66 – 67, Reiner), and

wherein each of said groups of virtual construct intervals covers a unique group of event values (Col.44, lines 1 – 2 and 7 – 10, Reiner), and wherein said inserting of said predicate values comprises inserting said predicate intervals only into said construct intervals that have corresponding event values (Col. 13, lines 55 – 57, Reiner<sup>11</sup>).

Regarding Claim 17, Reiner discloses a method, further comprising maintaining locations of said predicate intervals within said groups of virtual construct intervals using a predicate ID bitmap vector (Col. 63, lines 36 – 38, Reiner<sup>12</sup>).

Regarding Claim 18, Reiner discloses a method, wherein said process of defining said groups of virtual construct intervals comprises beginning all virtual construct intervals in a group of virtual construct intervals at the same attribute value

---

<sup>11</sup> Wherein examiner interprets the step of appending partition-matching predicates as a step of inserting predicate intervals into the same sized virtual construct claimed. In other words, matching the partitions implies matching the size.

<sup>12</sup> Examiner interprets the pointer to the bitmap as an ID bitmap vector claimed. Reiner further discloses that this pointer is used to indicate which buffers are full. This implies the step of locating.

(Col. 10, lines 50 – 55, Reiner<sup>13</sup>) and ending said virtual construct intervals in said group of virtual construct intervals at different attribute values (Col. 9, lines 17 – 28, Reiner<sup>14</sup>).

Regarding Claim 19, Reiner discloses a method, wherein all of said groups of said virtual construct intervals within said query index have the same pattern of different sized of virtual construct intervals (Col. 4 and 25, lines 34 – 37 and 39 – 40; respectively, Reiner).

Regarding Claim 20, Reiner discloses a method, wherein said process of inserting said predicate intervals into said virtual construct intervals, comprises inserting said predicate interval into the same sized virtual construct interval (Col. 13, lines 55 – 57, Reiner<sup>15</sup>).

Regarding Claim 22, Reiner discloses a method, wherein event values of said predicate intervals are aligned with the same event values of said virtual construct intervals, such that the same event values of multiple predicate intervals are inserted

---

<sup>13</sup> Wherein same key value corresponds to same attribute value claimed. In addition, the step of simultaneously indexing in accord to the same key value corresponds to beginning the predicates at the same attribute value claimed.

<sup>14</sup> The two queries, before insertion and after insertion of predicates (disclosed by Reiner), begin with the same attribute values, such as, "SELECT name". Regarding the same citation, Reiner discloses ending the two queries at different attribute values, such as, "WHERE department" and "employee.rowid<0.0.2"

<sup>15</sup> Wherein examiner interprets the step of appending partition-matching predicates as a step of inserting predicate intervals into the same sized virtual construct claimed. In other words, matching the partitions implies matching the size.

into the same event value within virtual construct interval (Col. 9, lines 36 – 39, Reiner<sup>16</sup>).

Regarding Claim 23, Reiner discloses a method for maintaining and using a query index, wherein queries within said query index have predicate intervals, said method comprising:

defining groups of virtual construct intervals (Col. 2 and 8, lines 65 – 67 and 37 – 40; respectively, Reiner);

inserting each of said predicate intervals into at least one of said groups of said virtual construct intervals (Col. 9, lines 12 – 13 and 27 – 28, Reiner),

wherein each of said groups of virtual construct intervals is adapted to hold multiple predicate intervals (Col. 2, lines 66 – 67, Reiner), and wherein each of said groups of virtual construct intervals covers a unique group of event values (Col. 44, lines 1 – 2 and 7 – 10, Reiner),

wherein said inserting of said predicate values comprises inserting said predicate intervals only into said construct intervals that have corresponding event values (Col. 13, lines 55 – 57, Reiner<sup>17</sup>), and

wherein said defining process only defines virtual construct intervals that are between the minimum and maximum possible attribute values of said predicate intervals (Col. 9 – 10 and 13, lines 67 and 1 – 2, and 43 – 45; respectively, Reiner) .

---

<sup>16</sup> Wherein examiner interprets the step of aligning with the same event values as the step of appending a predicate for matching records claimed.

Regarding Claim 24, Reiner discloses a method, further comprising maintaining locations of said predicate intervals within said groups of virtual construct intervals using a predicate ID bitmap vector (Col. 63, lines 36 – 38, Reiner<sup>18</sup>).

Regarding Claim 25, Reiner discloses a method, wherein said process of defining said groups of virtual construct intervals comprises beginning all virtual construct intervals in a group of virtual construct intervals at the same attribute value (Col. 10, lines 50 – 55, Reiner<sup>19</sup>) and ending said virtual construct intervals in said group of virtual construct intervals at different attribute values (Col. 9, lines 17 – 28, Reiner<sup>20</sup>).

Regarding Claim 26, Reiner discloses a method, wherein all of said groups of said virtual construct intervals within said query index have the same pattern of different sized of virtual construct intervals (Col. 4 and 25, lines 34 – 37 and 39 – 40; respectively, Reiner).

---

<sup>17</sup> Wherein examiner interprets the step of appending partition-matching predicates as a step of inserting predicate intervals into the same sized virtual construct claimed. In other words, matching the partitions implies matching the size.

<sup>18</sup> Examiner interprets the pointer to the bitmap as an ID bitmap vector claimed. Reiner further discloses that this pointer is used to indicate which buffers are full. This implies the step of locating.

<sup>19</sup> Wherein same key value corresponds to same attribute value claimed. In addition, the step of simultaneously indexing in accord to the same key value corresponds to beginning the predicates at the same attribute value claimed.

<sup>20</sup> The two queries, before insertion and after insertion of predicates (disclosed by Reiner), begin with the same attribute values, such as, "SELECT name". Regarding the same citation, Reiner discloses ending the two queries at different attribute values, such as, "WHERE department" and "employee.rowid<0.0.2"

Regarding Claim 27, Reiner discloses a method, wherein said process of inserting said predicate intervals into said virtual construct intervals, comprises inserting said predicate interval into the same sized virtual construct interval (Col. 13, lines 55 – 57, Reiner<sup>21</sup>).

Regarding Claim 29, Reiner discloses a method, wherein event values of said predicate intervals are aligned with the same event values of said virtual construct intervals, such that the same event values of multiple predicate intervals are inserted into the same event value within virtual construct interval (Col. 9, lines 36 – 39, Reiner<sup>22</sup>).

Regarding Claim 30, Reiner discloses a service adapted to maintains and use a query index, wherein queries within said query index have predicate intervals, said service:

defining groups of virtual construct intervals (Col. 2 and 8, lines 65 – 67 and 37 – 40; respectively, Reiner); and

inserting each of said predicate intervals into at least one of said groups of said virtual construct intervals (Col. 9, lines 12 – 13 and 27 – 28, Reiner),

wherein each of said groups of virtual construct intervals is adapted to hold multiple predicate intervals (Col. 2, lines 66 – 67, Reiner), and

---

<sup>21</sup> Wherein examiner interprets the step of appending partition-matching predicates as a step of inserting predicate intervals into the same sized virtual construct claimed. In other words, matching the partitions implies matching the size.

wherein each of said groups of virtual construct intervals covers a unique group of event values (Col. 44, lines 1 – 2 and 7 – 10, Reiner), and wherein said inserting of said predicate values comprises inserting said predicate intervals only into said construct intervals that have corresponding event values (Col. 13, lines 55 – 57, Reiner<sup>23</sup>).

Regarding Claim 31, Reiner discloses a service, further comprising maintaining locations of said predicate intervals within said groups of virtual construct intervals using a predicate ID bitmap vector (Col. 63, lines 36 – 38, Reiner<sup>24</sup>).

Regarding Claim 32, Reiner discloses a service, wherein said process of defining said groups of virtual construct intervals comprises beginning all virtual construct intervals in a group of virtual construct intervals at the same attribute value (Col. 10, lines 50 – 55, Reiner<sup>25</sup>) and ending said virtual construct intervals in said group of virtual construct intervals at different attribute values (Col. 9, lines 17 – 28, Reiner<sup>26</sup>).

---

<sup>22</sup> Wherein examiner interprets the step of aligning with the same event values as the step of appending a predicate for matching records claimed.

<sup>23</sup> Wherein examiner interprets the step of appending partition-matching predicates as a step of inserting predicate intervals into the same sized virtual construct claimed. In other words, matching the partitions implies matching the size.

<sup>24</sup> Examiner interprets the pointer to the bitmap as an ID bitmap vector claimed. Reiner further discloses that this pointer is used to indicate which buffers are full. This implies the step of locating.

<sup>25</sup> Wherein same key value corresponds to same attribute value claimed. In addition, the step of simultaneously indexing in accord to the same key value corresponds to beginning the predicates at the same attribute value claimed.

<sup>26</sup> The two queries, before insertion and after insertion of predicates (disclosed by Reiner), begin with the same attribute values, such as, "SELECT name". Regarding the same citation, Reiner discloses ending the two queries at different attribute values, such as, "WHERE department" and "employee.rowid<0.0.2"

Regarding Claim 33, Reiner discloses a service, wherein all of said groups of said virtual construct intervals within said query index have the same pattern of different sized of virtual construct intervals (Col. 4 and 25, lines 34 – 37 and 39 – 40; respectively, Reiner).

Regarding Claim 34, Reiner discloses a service, wherein said process of inserting said predicate intervals into said virtual construct intervals, comprises inserting said predicate interval into the same sized virtual construct interval (Col. 13, lines 55 – 57, Reiner<sup>27</sup>).

Regarding Claim 36, Reiner discloses a service, wherein event values of said predicate intervals are aligned with the same event values of said virtual construct intervals, such that the same event values of multiple predicate intervals are inserted into the same event value within virtual construct interval (Col. 9, lines 36 – 39, Reiner<sup>28</sup>).

Regarding Claim 37, Reiner discloses a system for maintaining and using a query index, wherein queries within said query index have predicate intervals, said system comprising:

a plurality of bitmap vectors which define groups of virtual construct intervals;

---

<sup>27</sup> Wherein examiner interprets the step of appending partition-matching predicates as a step of inserting predicate intervals into the same sized virtual construct claimed. In other words, matching the partitions implies matching the size.

a predicate insertion handler adapted to insert each of said predicate intervals into at least one of said groups of said virtual construct intervals (Col. 9, lines 12 – 13 and 27 – 28, Reiner),

wherein each of said groups of virtual construct intervals is adapted to hold multiple predicate intervals (Col. 2, lines 66 – 67, Reiner), and

wherein each of said groups of virtual construct intervals covers a unique group of event values (Col. 44, lines 1 – 2 and 7 – 10, Reiner), and wherein said inserting of said predicate values comprises inserting said predicate intervals only into said construct intervals that have corresponding event values (Col. 13, lines 55 – 57, Reiner<sup>29</sup>).

Regarding Claim 38, Reiner discloses a system, further comprising a predicate ID bitmap vector adapted to maintain locations of said predicate intervals within said groups of virtual construct intervals (Col. 63, lines 36 – 38, Reiner<sup>30</sup>).

Regarding Claim 39, Reiner discloses a system, wherein all virtual construct intervals in a group of virtual construct intervals begin at the same attribute value (Col.

---

<sup>28</sup> Wherein examiner interprets the step of aligning with the same event values as the step of appending a predicate for matching records claimed.

<sup>29</sup> Wherein examiner interprets the step of appending partition-matching predicates as a step of inserting predicate intervals into the same sized virtual construct claimed. In other words, matching the partitions implies matching the size.

<sup>30</sup> Examiner interprets the pointer to the bitmap as an ID bitmap vector claimed. Reiner further discloses that this pointer is used to indicate which buffers are full. This implies the step of locating.



Art Unit: 2162

10, lines 50 – 55, Reiner<sup>31</sup>) and end at different attribute values (Col. 9, lines 17 – 28, Reiner<sup>32</sup>).

Regarding Claim 40, Reiner discloses a system, wherein all of said groups of said virtual construct intervals have the same pattern of different sized of virtual construct intervals (Col. 4 and 25, lines 34 – 37 and 39 – 40; respectively, Reiner).

Regarding Claim 41, Reiner discloses a system, wherein said predicate insertion handler inserts said predicate intervals into the same sized virtual construct intervals (Col. 13, lines 55 – 57, Reiner<sup>33</sup>).

Regarding Claim 43, Reiner discloses a system, wherein event values of said predicate intervals are aligned with the same event values of said virtual construct intervals, such that the same event values of multiple predicate intervals are inserted into the same event value within virtual construct interval (Col. 9, lines 36 – 39, Reiner<sup>34</sup>).

---

<sup>31</sup> Wherein same key value corresponds to same attribute value claimed. In addition, the step of simultaneously indexing in accord to the same key value corresponds to beginning the predicates at the same attribute value claimed.

<sup>32</sup> The two queries, before insertion and after insertion of predicates (disclosed by Reiner), begin with the same attribute values, such as, "SELECT name". Regarding the same citation, Reiner discloses ending the two queries at different attribute values, such as, "WHERE department" and "employee.rowid<0.0.2"

<sup>33</sup> Wherein examiner interprets the step of appending partition-matching predicates as a step of inserting predicate intervals into the same sized virtual construct claimed. In other words, matching the partitions implies matching the size.

<sup>34</sup> Wherein examiner interprets the step of aligning with the same event values as the step of appending a predicate for matching records claimed.

Regarding Claim 44, Reiner discloses a program storage device readable by machine, tangibly embodying a program of instructions executable by the machine to perform a method for maintaining and using a query index, wherein queries within said query index have predicate intervals, said method comprising:

defining groups of virtual construct intervals (Col. 2 and 8, lines 65 – 67 and 37 – 40; respectively, Reiner); and

inserting each of said predicate intervals into at least one of said groups of said virtual construct intervals (Col. 9, lines 12 – 13 and 27 – 28, Reiner),

wherein each of said groups of virtual construct intervals is adapted to hold multiple predicate intervals (Col. 2, lines 66 – 67, Reiner), and

wherein said groups of virtual construct intervals have uniform lengths (Col. 25, lines 44 – 47, Reiner), and

wherein said predicate intervals have non-uniform lengths (Col. 25, lines 39 – 40, Reiner).

Regarding Claim 45, Reiner discloses a program storage device, wherein said method further comprises maintaining locations of said predicate intervals within said groups of virtual construct intervals using a predicate ID bitmap vector (Col. 63, lines 36 – 38, Reiner<sup>35</sup>).

---

<sup>35</sup> Examiner interprets the pointer to the bitmap as an ID bitmap vector claimed. Reiner further discloses that this pointer is used to indicate which buffers are full. This implies the step of locating.

Regarding Claim 46, Reiner discloses a program storage device, wherein said process of defining said groups of virtual construct intervals comprises beginning all virtual construct intervals in a group of virtual construct intervals at the same attribute value (Col. 10, lines 50 – 55, Reiner<sup>36</sup>) and ending said virtual construct intervals in said group of virtual construct intervals at different attribute values (Col. 9, lines 17 – 28, Reiner<sup>37</sup>).

Regarding Claim 47, Reiner discloses a program storage device, wherein all of said groups of said virtual construct intervals within said query index have the same pattern of different sized of virtual construct intervals (Col. 4 and 25, lines 34 – 37 and 39 – 40; respectively, Reiner).

Regarding Claim 48, Reiner discloses a program storage device, wherein said process of inserting said predicate intervals into said virtual construct intervals, comprises inserting said predicate interval into the same sized virtual construct interval (Col. 13, lines 55 – 57, Reiner<sup>38</sup>).

---

<sup>36</sup> Wherein same key value corresponds to same attribute value claimed. In addition, the step of simultaneously indexing in accord to the same key value corresponds to beginning the predicates at the same attribute value claimed.

<sup>37</sup> The two queries, before insertion and after insertion of predicates (disclosed by Reiner), begin with the same attribute values, such as, "SELECT name". Regarding the same citation, Reiner discloses ending the two queries at different attribute values, such as, "WHERE department" and "employee.rowid<0.0.2"

<sup>38</sup> Wherein examiner interprets the step of appending partition-matching predicates as a step of inserting predicate intervals into the same sized virtual construct claimed. In other words, matching the partitions implies matching the size.

Regarding Claim 50, Reiner discloses a program storage device, wherein event values of said predicate intervals are aligned with the same event values of said virtual construct intervals, such that the same event values of multiple predicate intervals are inserted into the same event value within virtual construct interval (Col. 9, lines 36 – 39, Reiner<sup>39</sup>).

***Claim Rejections - 35 USC § 103***

7. The following is a quotation of 35 U.S.C. 103(a) which forms the basis for all obviousness rejections set forth in this Office action:

(a) A patent may not be obtained though the invention is not identically disclosed or described as set forth in section 102 of this title, if the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been obvious at the time the invention was made to a person having ordinary skill in the art to which said subject matter pertains. Patentability shall not be negated by the manner in which the invention was made.

8. This application currently names joint inventors. In considering patentability of the claims under 35 U.S.C. 103(a), the examiner presumes that the subject matter of the various claims was commonly owned at the time any inventions covered therein were made absent any evidence to the contrary. Applicant is advised of the obligation under 37 CFR 1.56 to point out the inventor and invention dates of each claim that was not commonly owned at the time a later invention was made in order for the examiner to

---

<sup>39</sup> Wherein examiner interprets the step of aligning with the same event values as the step of appending a

consider the applicability of 35 U.S.C. 103(c) and potential 35 U.S.C. 102(e), (f) or (g) prior art under 35 U.S.C. 103(a).

9. Claim 7, 14, 21, 28, 35, 42, and 49 are rejected under 35 U.S.C. 103(a) as being unpatentable over Reiner et al. (Reiner hereinafter) (US Patent No. 6,289,334 B1, issued: September 11, 2001) in view of Jagadish et al. (Jagadish hereinafter) (US Patent No. 7,010,522 B1, filed: June 17, 2002).

Regarding Claim 7, Reiner discloses all the limitations as disclosed above including length of predicate intervals (Col. 64, lines 32 – 35, Reiner) and inserting portions of predicate intervals into virtual construct intervals (Col. 9, lines 12 – 13 and 27 – 28, Reiner). However, Reiner is silent with respect to a method that determines if a predicate is larger than any of the virtual construct intervals. On the other hand, Jagadish discloses a method for inserting predicate intervals that comprises: inserting an initial portion of said predicate interval into the largest available virtual construct interval (Col. 5, lines 22 – 27 and 37 – 39, Jagadish<sup>40</sup>), wherein excess length of said predicate interval comprises a remnant predicate interval (Col. 5, lines 42 – 43, Jagadish<sup>41</sup>); and inserting the remnant predicate interval into the same length virtual construct interval (Col. 5, lines 32 – 34 and 37 – 39, Jagadish). It would have been obvious to one of ordinary skill in the art at the time the invention was made to incorporate the Jagadish's teachings to the system of Reiner. Skilled artisan would have

---

predicate for matching records claimed.

<sup>40</sup> Examiner interprets the chopped Q pieces, disclosed by Jagadish, as the portions of predicate intervals claimed.

been motivated to do so, as suggested by Jagadish (Col. 2, lines 6 –7 and 29 – 31, Jagadish), to be able to efficiently find all strings approximately containing a given query substring from a large collection of strings by using all (or part) of overlapping pieces. In addition, both of the references (Reiner and Jagadish) teach features that are directed to analogous art and they are directed to the same field of endeavor, such as, databases management systems and indexing queries. This close relation between both of the references highly suggests an expectation of success.

Regarding Claim 14, the combination of Reiner in view of Jagadish discloses a method, wherein if a predicate interval is larger than any of said virtual construct intervals, said inserting process comprises:

inserting an initial portion of said predicate interval into the largest available virtual construct interval (Col. 9, lines 12 – 13 and 27 – 28, Reiner; and Col. 5, lines 22 – 27 and 37 – 39, Jagadish<sup>42</sup>), wherein excess length of said predicate interval comprises a remnant predicate interval (Col. 5, lines 42 – 43, Jagadish<sup>43</sup>); and

inserting the remnant predicate interval into the same length virtual construct interval (Col. 5, lines 32 – 34 and 37 – 39, Jagadish).

---

<sup>41</sup> The step of truncating the piece of query of length longer than q, implies that there is excess length of the predicates.

<sup>42</sup> Examiner interprets the chopped Q pieces, disclosed by Jagadish, as the portions of predicate intervals claimed.

<sup>43</sup> The step of truncating the piece of query of length longer than q, implies that there is excess length of the predicates.

Regarding Claim 21, the combination of Reiner in view of Jagadish discloses a method, wherein if a predicate interval is larger than any of said virtual construct intervals, said inserting process comprises:

inserting an initial portion of said predicate interval into the largest available virtual construct interval (Col. 9, lines 12 – 13 and 27 – 28, Reiner; and Col. 5, lines 22 – 27 and 37 – 39, Jagadish<sup>44</sup>), wherein excess length of said predicate interval comprises a remnant predicate interval (Col. 5, lines 42 – 43, Jagadish<sup>45</sup>); and

inserting the remnant predicate interval into the same length virtual construct interval (Col. 5, lines 32 – 34 and 37 – 39, Jagadish).

Regarding Claim 28, the combination of Reiner in view of Jagadish discloses a method, wherein if a predicate interval is larger than any of said virtual construct intervals, said inserting process comprises:

inserting an initial portion of said predicate interval into the largest available virtual construct interval (Col. 9, lines 12 – 13 and 27 – 28, Reiner; and Col. 5, lines 22 – 27 and 37 – 39, Jagadish<sup>46</sup>), wherein excess length of said predicate interval comprises a remnant predicate interval (Col. 5, lines 42 – 43, Jagadish<sup>47</sup>); and

---

<sup>44</sup> Examiner interprets the chopped Q pieces, disclosed by Jagadish, as the portions of predicate intervals claimed.

<sup>45</sup> The step of truncating the piece of query of length longer than q, implies that there is excess length of the predicates.

<sup>46</sup> Examiner interprets the chopped Q pieces, disclosed by Jagadish, as the portions of predicate intervals claimed.

<sup>47</sup> The step of truncating the piece of query of length longer than q, implies that there is excess length of the predicates.

inserting the remnant predicate interval into the same length virtual construct interval (Col. 5, lines 32 – 34 and 37 – 39, Jagadish).

Regarding Claim 35, the combination of Reiner in view of Jagadish discloses a service, wherein if a predicate interval is larger than any of said virtual construct intervals, said inserting process comprises:

inserting an initial portion of said predicate interval into the largest available virtual construct interval (Col. 9, lines 12 – 13 and 27 – 28, Reiner; and Col. 5, lines 22 – 27 and 37 – 39, Jagadish<sup>48</sup>), wherein excess length of said predicate interval comprises a remnant predicate interval (Col. 5, lines 42 – 43, Jagadish<sup>49</sup>); and

inserting the remnant predicate interval into the same length virtual construct interval (Col. 5, lines 32 – 34 and 37 – 39, Jagadish).

Regarding Claim 42, the combination of Reiner in view of Jagadish discloses a system, wherein if a predicate interval is larger than any of said virtual construct intervals, said predicate insertion handler:

inserts an initial portion of said predicate interval into the largest available virtual construct interval (Col. 9, lines 12 – 13 and 27 – 28, Reiner; and Col. 5, lines 22 – 27

---

<sup>48</sup> Examiner interprets the chopped Q pieces, disclosed by Jagadish, as the portions of predicate intervals claimed.

<sup>49</sup> The step of truncating the piece of query of length longer than q, implies that there is excess length of the predicates.



Art Unit: 2162

and 37 – 39, Jagadish<sup>50</sup>), wherein excess length of said predicate interval comprises a remnant predicate interval (Col. 5, lines 42 – 43, Jagadish<sup>51</sup>); and

inserts the remnant predicate interval into the same length virtual construct interval (Col. 5, lines 32 – 34 and 37 – 39, Jagadish).

Regarding Claim 49, the combination of Reiner in view of Jagadish discloses a program storage device, wherein if a predicate interval is larger than any of said virtual construct intervals, said inserting process comprises:

inserting an initial portion of said predicate interval into the largest available virtual construct interval (Col. 9, lines 12 – 13 and 27 – 28, Reiner; and Col. 5, lines 22 – 27 and 37 – 39, Jagadish<sup>52</sup>), wherein excess length of said predicate interval comprises a remnant predicate interval (Col. 5, lines 42 – 43, Jagadish<sup>53</sup>); and

inserting the remnant predicate interval into the same length virtual construct interval (Col. 5, lines 32 – 34 and 37 – 39, Jagadish).

---

<sup>50</sup> Examiner interprets the chopped Q pieces, disclosed by Jagadish, as the portions of predicate intervals claimed.

<sup>51</sup> The step of truncating the piece of query of length longer than q, implies that there is excess length of the predicates.

<sup>52</sup> Examiner interprets the chopped Q pieces, disclosed by Jagadish, as the portions of predicate intervals claimed.

<sup>53</sup> The step of truncating the piece of query of length longer than q, implies that there is excess length of the predicates.

***Prior Art Made Of Record***

1. Reiner et al. (US Patent No. 6,289,334 B1, issued: September 11, 2001) discloses an apparatus and method for recomposing database queries for database management system including multiprocessor digital data processing system.
2. Jagadish et al. (US Patent No. 7,010,522 B1, filed: June 17, 2002) discloses a method of performing approximate substring indexing.
3. Antoshenkov (US Patent No. 5,664,172) discloses a range-based query optimizer.

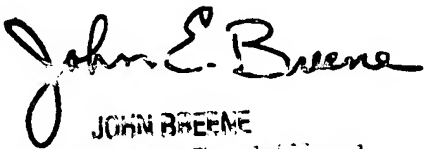
***Points of Contact***

Any inquiry concerning this communication or earlier communications from the examiner should be directed to Giovanna Colan whose telephone number is (571) 272-2752. The examiner can normally be reached on 8:30 am - 5:00 pm.

If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, John Breene can be reached on (571) 272-4107. The fax phone number for the organization where this application or proceeding is assigned is 571-273-8300.

Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see <http://pair-direct.uspto.gov>. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free).

Giovanna Colan  
Examiner  
Art Unit 2162  
March 22, 2006

  
JOHN BREENE  
SUPERVISORY PATENT EXAMINER  
TECHNOLOGY CENTER 2100

5A